

Experimental Algorithm for the Maximum Independent Set Problem

Anatoly D. Plotnikov*

Abstract

We develop an experimental algorithm for the exact solving of the maximum independent set problem. The algorithm consecutively finds the maximal independent sets of vertices in an arbitrary undirected graph such that the next such set contains more elements than the preceding one. For this purpose, we use a technique, developed by Ford and Fulkerson for the finite partially ordered sets, in particular, their method for partition of a poset into the minimum number of chains with finding the maximum antichain. In the process of solving, a special digraph is constructed, and a conjecture is formulated concerning properties of such digraph. This allows to offer of the solution algorithm. Its theoretical estimation of running time equals to is $O(n^8)$, where n is the number of graph vertices. The offered algorithm was tested by a program on random graphs. The testing the confirms correctness of the algorithm.

MSC 2000: 05C85, 68Q17.

KEYWORDS: the maximum independent set, a clique, NP-hard, NP-complete, the class NP, a polynomial-time algorithm, the partially ordered set.

1 Introduction

A set of all undirected n -vertex graphs without loops and multiple edges is denoted by L_n .

Let there be a graph $G = (V, \Gamma) \in L_n$, where V is the set of graph vertices, and Γ is a mapping from V to V . Any subgraph $Q_1 = (V_1, \Gamma_1)$ of G is called a *clique* induced by V_1 , if

$$\Gamma_1(v) = V_1 \setminus \{v\}$$

for all $v \in V_1$. In a special case, when $Card(V_1) = 1$, the one-vertex subgraph $Q_1 = (V_1, \Gamma_1)$ is called a *single-vertex clique*.

A clique Q_1 is called *maximal* if any vertex $v \in V$ cannot be attached to it so that the new vertex set also has formed a clique of the graph G . A clique \hat{Q} is called *maximum* if the graph G has not a clique of the greater size than \hat{Q} .

*E-mail address: `akplokt(at)kvnk.ua` (remove k's)

It be required to find the maximum clique of a graph $G \in L_n$. The known problem of finding the maximum clique has formulated [3].

A vertex set $U \subseteq V$ of G is called *independent* if

$$U \cap \Gamma(U) = \emptyset.$$

An independent set U of graph vertices is called *maximal (MIS)* if

$$U \cup \Gamma(U) = V.$$

A MIS \tilde{U} is called *maximum (MMIS)* if $Card(\tilde{U}) \geq Card(U)$ for any MIS U of G .

It be required to find the MMIS of a graph G . Again, we have formulated well-know the maximum independent set problem (MISP).

Both of the formulated above problems are NP-complete [3, 1]. They are closely connected with each other: the solution of one of them attracts the solution another.

A graph \bar{G} is called *complement* to the graph G if it has the same vertex set, and edges join two vertices of the graph \bar{G} iff these vertices are non-adjacent in G .

It is not difficult to see that any clique of G corresponds to the independent set of graph vertices in \bar{G} , and conversely. Therefore, the finding the maximum independent set in one graph is equivalent to the finding the maximum clique in its complement graph. In the given work, we examine the maximum independent set problem.

The intention of the given paper is to design a polynomial-time algorithm for the exact solving of the maximum independent set problem in an arbitrary undirected graph. For this purpose, we use a technique, developed by Ford and Fulkerson (see [2] and Appendix A) for the finite partially ordered sets, in particular, their method of partition poset into the minimum number of chains with finding the maximum antichain. In the process of the solving, a special digraph is constructed, and a conjecture is formulated concerning properties such digraph. This allows to offer a solution algorithm, a theoretical estimation of running time for which equals to is $O(n^8)$, where n is the number of graph vertices. The offered algorithm was tested by a program. The testing confirms the correctness of the algorithm.

Notice that the author did not aspire to creation of the most optimum and fast algorithm. In work [8], Tarjan has presented a table, which shows a dynamics of perfecting a complexity evaluation for solving some problems. Whence one can make a conclusion that after the appearance of an initial algorithm for solving certain problem, its improvement is found quickly. It is presented that proposed algorithm also can be improved hereinafter.

2 The Basic Definitions

Let there is a graph $G = (V, \Gamma) \in L_n$. We will partition the vertex set V into subsets

$$V^0, V^1, \dots, V^m \quad (1)$$

in such a way that a subset V^k ($k = 0, 1, \dots, m$) is a MIS of a subgraph $G_k = (V \setminus (V^0 \cup V^1 \cup \dots \cup V^{k-1}), \Gamma_k) = (V^k \cup \dots \cup V^m, \Gamma_k)$. Clearly, $G_0 = (V^0 \cup \dots \cup V^m, \Gamma_0) = G$.

By the given undirected graph G and the partition (1) we can construct a digraph $\vec{G}(V^0) = (V, \vec{\Gamma})$ in the following way. If an edge of G joins a vertex $v_i \in V^{k_1}$ with a vertex $v_j \in V^{k_2}$ then this edge is replaced by an arc (v_i, v_j) when $k_1 < k_2$. The vertex v_i is called the *tail* of (v_i, v_j) , and the vertex v_j is called the *head* of this arc.

As the result we have an acyclic digraph $\vec{G}(V^0) = (V, \vec{\Gamma})$. The set V^0 is called *initiating*.

In general case we can construct a set $D(G)$ of different acyclic digraphs as it was indicated above. Each digraph of $D(G)$ corresponds to the graph $G \in L_n$. Further we will consider *only* digraphs of $D(G)$.

The maximum length $\rho(v)$ of a directed path, connecting a vertex $v \in V$ with some vertex of the initiating set V^0 , is called the *rank* of v . The set of all graph vertices having the same rank $\rho(v) = k$ is called the *k-th layer* of $\vec{G}(V^0)$ and designated as V^k .

To apply the partially ordered set technique, each digraph $\vec{G}(V^0)$ is assigned to a transitive closure graph (TCG) $\vec{G}_t(V^0) = (V, \vec{\Gamma}_t)$ [7, 9]. As the digraph $\vec{G}(V^0)$ is acyclic and loopless, its transitive closure $\vec{G}_t(V^0)$ is a graph of a strict partial order $(V, >)$. Further, we will not distinguish the transitive closure graph $\vec{G}_t(V^0)$ and partially ordered set (poset) $(V, >)$. Therefore, we will consider, for example, antichains of the TCG $\vec{G}_t(V^0)$.

There exists an efficient algorithm to construct the TCG. Its running time is equal to $O(n^3)$ (see, for example, [7, 6]).

An arc (v_i, v_j) of $\vec{G}_t(V^0)$ will be called *essential* if there exists the arc (v_i, v_j) of the digraph $\vec{G}(V^0)$. Otherwise, the arc (v_i, v_j) will be called *fictitious*. An essential arc is also designated as $v_i > v_j$, and a fictitious arc is also designated as $v_i \gg v_j$. Obviously that any fictitious arc of $\vec{G}_t(V^0)$ determines two independent vertices of the digraph $\vec{G}(V^0)$.

Let there is a poset (A, \geq) .

If $a \geq b$ or $b \geq a$, the elements a and b of A are called *comparable*. If $a \not\geq b$ and $b \not\geq a$, such pair of elements is called *incomparable*.

If $A_1 \subseteq A$ and each pair of elements of A_1 is comparable, we shall say that A_1 determines a *chain* of (A, \geq) . If $A_1 \subseteq A$ and each pair of elements of A_1 is incomparable, we shall say that A_1 be an *antichain* of (A, \geq) . The antichain A_1 is *the maximum* in (A, \geq) , if $\text{Card}(A_1) \geq \text{Card}(A^*)$ for any antichain $A^* \subseteq A$ in (A, \geq) .

We say that poset (A, \geq) is partitioned into chains A_1, \dots, A_p , if each A_i ($A_i \neq \emptyset, i = \overline{1, p}$) be a chain,

$$\bigcup_{i=1}^p A_i = A,$$

and $A_i \cap A_j = \emptyset$, when $i \neq j$ ($i, j \in \{1, \dots, p\}$).

The partition of the poset (A, \geq) into chains is called *minimum*, if it has the minimum number of elements p in comparison with other partitions of (A, \geq) into chains. Such partition also is called *minimum chain partition (MCP)* of poset (A, \geq) .

As $\vec{G}_t(V^0)$ is a graph of strict partial order, we can find MCP $\mathcal{P} = \{S_1, \dots, S_p\}$. In common case, this partition is ambiguous.

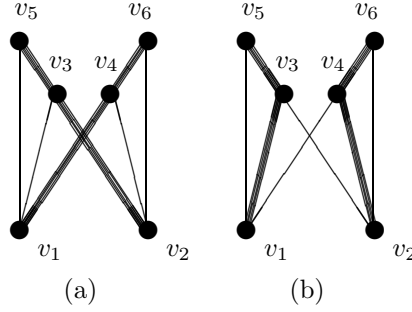


Figure 1: Different MCPs of the transitive closure graph

Different MCPs of the transitive closure graph is shown on the Fig. 1 (a) and (b). The digraph arcs, belonging chains of MCP, are represented by thick lines. Here and hereinafter, we suppose that orientation of arcs of the digraph is from below to upwards.

Let $V(S_q)$ ($q = \overline{1, p}$) be the set of vertices, belonging to the chain S_q . If vertices $v_i, v_j \in V(S_q)$ are endpoints for a fictitious arc $v_i \gg v_j$ then the vertex v_j is called *marked*. The set of all marked vertices of $\vec{G}_t(V^0)$ is determined by the found MCP \mathcal{P} and it differs for different MCPs. The set of all marked vertices of $\vec{G}_t(V^0)$ is designated as $\mathcal{B}(\mathcal{P})$.

For example, for the MCP \mathcal{P}_1 represented in Fig. 1 (a), we have $\mathcal{B}(\mathcal{P}_1) = \{v_5, v_6\}$, and for the MCP \mathcal{P}_2 represented in Fig. 1 (b), we have $\mathcal{B}(\mathcal{P}_2) = \emptyset$.

Lemma 1 *Let $\vec{G}(V^0)$ be a digraph, and $\vec{G}_t(V^0)$ be its transitive closure graph. If $\mathcal{B}(\mathcal{P}) = \emptyset$, where \mathcal{P} is an MCP of $\vec{G}_t(V^0)$, then the maximum antichain \mathcal{U} is the MMIS of the graph $G \in L_n$, and the MCP determines the minimum clique partition of G .*

If conditions of Lemma 1 are satisfied then each chain $S_q \in \mathcal{P}$ is a clique of $\vec{G}(V^0)$. Therefore, the MCP \mathcal{P} is the minimum clique partition.

On the other hand, vertices of the maximum antichain \mathcal{U} of $\vec{G}_t(V^0)$ belong to distinct cliques, that is, the number of vertices in the MMIS is equal to the number of vertices in the set \mathcal{U} . Q.E.D.

Notice that Lemma 1 may be satisfied if the digraph $\vec{G}(V^0)$ has no transitive orientation.

It is obvious that any antichain $U \subset V$ of $\vec{G}_t(V^0)$ determines an independent vertex set of the digraph $\vec{G}(V^0)$, and an independent vertex set $U \subset V$ of $\vec{G}(V^0)$ determines an antichain of $\vec{G}_t(V^0)$ if and only if no two vertices of U belong to the same directed chain of $\vec{G}(V^0)$.

3 A vertex-saturated digraph

Let there is an acyclic digraph $\vec{G}(V^0) = (V, \vec{\Gamma})$.

Further, let $W \subset V$ be some independent vertex set. For the digraph $\vec{G}(V^0)$ we define an unary operation *cutting* $\sigma_W(\vec{G}(V^0))$. This operation consists of reorientation of all arcs of $\vec{G}(V^0)$ incoming into vertices of the set W . It is easy to see that the result of this operation is also a digraph $\vec{G}(Y^0)$, where

$$Y^0 = (V^0 \setminus \vec{\Gamma}^{-1}(W)) \cup W.$$

Here, $\vec{\Gamma}^{-1}$ is a mapping, inverse to $\vec{\Gamma}$.

Theorem 1 *Let there be a digraph $\vec{G}(V^0) \in D(G)$ and W be some independent vertex set. Then a digraph $\vec{G}(Y^0) = \sigma_W(\vec{G}(V^0))$ is also acyclic and $\vec{G}(Y^0) \in D(G)$.*

Indeed, since the digraph $\vec{G}(V^0) = (V, \vec{\Gamma})$ is acyclic then any its part is also an acyclic digraph. Thus, a directed subgraph $\vec{G}_1 = (V \setminus W, \vec{\Gamma}_1)$ is acyclic.

Obviously, $\vec{G}_1 \subset \vec{G}(Y^0)$. Attach the independent vertex set W to the subgraph \vec{G}_1 . Join each vertex $x \in W$ with a vertex y of \vec{G}_1 by the arc (x, y) if and only if there exists the arc (y, x) of the digraph $\vec{G}(V^0)$. It is evident that the resulting digraph $\vec{G}(Y^0)$ is also acyclic.

At last, we have $\vec{G}(Y^0) \in D(G)$ since any reorientation of arcs of the digraph $\vec{G}(V^0)$ does not change independence relation of its vertices. Q.E.D.

Let there be a digraph $\vec{G}(V^0)$ and its transitive closure graph $\vec{G}_t(V^0)$.

We can find the MCP \mathcal{P} of the graph $\vec{G}_t(V^0)$ constructing the maximum antichain \mathcal{U} simultaneously as it is described in Appendix A.

In general case we can find some distinct maximum antichains.

We will say that an antichain U_1 *precedes* an antichain U_2 in the graph $\vec{G}_t(V^0)$ and designate it as $U_1 \prec U_2$ if for all vertices $x \in U_1 \setminus U_2$ there is a vertex $y \in U_2 \setminus U_1$ such that $x \leq y$.

By means of Ford and Fulkerson's methodology such a maximum antichain \mathcal{U} of the TCG $\vec{G}_t(V^0)$ can be found that precedes any other maximum antichain \mathcal{U}_1 , i.e. $\mathcal{U}_1 \prec \mathcal{U}$ for any antichain \mathcal{U}_1 of the TCG $\vec{G}_t(V^0)$. This antichain of the graph $\vec{G}_t(V^0)$ we will call *general*.

In addition to the general antichain, we may find other maximum antichains of $\vec{G}_t(V^0)$ if they exist. So for any vertex $v \in V$ of the TCG $\vec{G}_t(V^0)$, it is possible to find a maximum antichain $\mathcal{U}(v)$ such that $v \in \mathcal{U}(v)$. Technically, to find the antichain $\mathcal{U}(v)$, it is sufficient, in the adjacent matrix of $\vec{G}_t(V^0)$ containing the maximum number of units in allowable cells (and the marks are appointed by the Ford-Fulkerson's algorithm), to add the mark $(*)$ to the existing marks for a row, corresponding to vertex v , and to execute a cycle of appointment of marks. In this case, in the first step of appointment of marks, all columns are marked, which contain the admissible cells (including a chosen cell). Clearly, the antichain $\mathcal{U}(v)$ will be general for the vertex v , that is, any other antichain, containing vertex v , will precede this antichain.

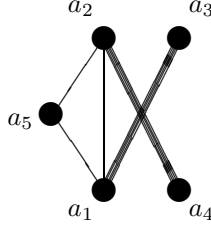


Figure 2: The partially ordered set

Notice that in general case it is not true that for any vertex $v \in V$ of the TCG $\vec{G}_t(V^0)$ there exists a maximum antichain $\mathcal{U}(v)$. For example, consider the digraph of the partially ordered set, shown in Fig. 2. It is easy to see that there are no maximum antichains $\mathcal{U}(a_1)$ and $\mathcal{U}(a_2)$ in it.

A directed subgraph $\vec{G}(V^k)$ ($k = \overline{0, m-1}$) of the digraph $\vec{G}(V^0)$ we will call *saturated with respect to the initiating set* V^k , if, in its transitive closure graph $\vec{G}_t(V^k)$, any maximum antichain $\mathcal{U}(v) \subset V$, when it exists, is a MIS of the subgraph $\vec{G}(V^k)$ and satisfies the relation: $Card(\mathcal{U}(v)) = Card(V^k)$. Evidently, the subgraph $\vec{G}(V^m)$ has no arcs, and therefore it is saturated with respect to its initiating set always.

A digraph $\vec{G}(V^0)$ is called *vertex-saturated (VS-digraph)* if any of its directed subgraphs $\vec{G}(V^k)$ ($k = \overline{0, m}$), induced by the layer V^k , is saturated with respect to the initiating set V^k .

Notice that digraph, represented in Fig. 1, is not vertex-saturated.

Let there be some digraph $\vec{G}(V^0)$. To construct a VS-digraph, we use the following algorithm.

The algorithm VS.

- Step 1.* Put $k := 0$ and $\alpha := \mathbf{false}$.
- Step 2.* Find the transitive closure graph $\vec{G}_t(V^k)$.
- Step 3.* Construct an MCP of $\vec{G}_t(V^k)$.
- Step 4.* Find the maximum antichain of $\vec{G}_t(V^k)$ for each vertex $v \in V^k \cup \dots \cup V^m$.
- Step 5.* Check whether each of the found maximum antichains $\mathcal{U}(v)$ of the graph $\vec{G}_t(V^k)$ is a MIS of the digraph $\vec{G}(V^k)$ and $\text{Card}(\mathcal{U}(v)) = \text{Card}(V^k)$. If it is true, finish the design of the digraph $\vec{G}(V^k)$ saturated with respect to the initiating set V^k . Go to Step 6.
- Otherwise complete the found antichain $\mathcal{U}(v)$ (when it is necessary) to a MIS, put $W := \mathcal{U}(v)$ and construct a new acyclic digraph $\vec{G}(V^k)$ by the cutting operation $\sigma_W(\vec{G}(V^0))$, put $\alpha := \mathbf{true}$ and construct the new digraph $\vec{G}(V^0)$. Return to Step 2.
- Step 6.* Compute $k := k + 1$. If $k < m$ then distinguish the transitive closure graph $\vec{G}_t(V^k)$ from $\vec{G}_t(V^{k-1})$, keeping all chains of the MCP of $\vec{G}_t(V^{k-1})$ which are incident to vertices of the new graph. Go to Step 4.
- If $k = m$ and $\alpha = \mathbf{true}$, go to Step 1. If $k = m$ and $\alpha = \mathbf{false}$, go to Step 7.
- Step 7.* Finish of this algorithm. A VS-digraph is constructed.

We will show that the algorithm VS constructs a vertex-saturated digraph.

Theorem 2 *Let $\vec{G}(V^0)$ be a digraph constructed by the algorithm VS. Next, let $\vec{G}(V^k) = (Y_k, \Gamma)$ ($k = \overline{0, m}$) be a directed subgraph induced by the layer V^k . Then each antichain $\mathcal{U} \subset Y_k \setminus V^k$ of the graph $\vec{G}_t(V^k) = (Y_k, \Gamma_t)$ obeys the following relation*

$$\text{Card}(V^k \cap \Gamma_t^{-1}\mathcal{U}) \geq \text{Card}(\mathcal{U}) \quad (2)$$

Assume that, in the directed subgraph $\vec{G}(V^k)$, an antichain $\mathcal{U} \subset Y_k \setminus V^k$ of $\vec{G}_t(V^k) = (Y_k, \Gamma_t)$ will be found such that

$$\text{Card}(V^k \cap \Gamma_t^{-1}(\mathcal{U})) < \text{Card}(\mathcal{U}).$$

Construct a set $\mathcal{U}^* = (V^k \setminus \Gamma_t^{-1}(\mathcal{U})) \cup \mathcal{U}$. Clearly, the set \mathcal{U}^* is independent in the directed subdigraph $\vec{G}(V^k)$ and $\text{Card}(\mathcal{U}^*) > \text{Card}(V^k)$.

Since the set V^k is an antichain of the TCG $\vec{G}_t(V^k) = (Y_k, \Gamma_t)$ then the set $V^k \setminus \Gamma_t^{-1}(\mathcal{U}) \subset V^k$ is an antichain of this graph. The set $\mathcal{U} \subset Y_k \setminus V^k$ is an antichain of $\vec{G}_t(V^k)$ by conditions of Theorem 2.

Obviously, $\mathcal{U} \cap \Gamma_t(V^k \setminus \Gamma_t^{-1}(\mathcal{U})) = \emptyset$, therefore the set \mathcal{U}^* is an antichain of $\vec{G}_t(V^k)$.

We have obtained a contradiction since the directed subgraph $\vec{G}(V^k)$ is constructed by the algorithm VS and the antichain \mathcal{U}^* could be discovered in Step 5. This proves the validity of Theorem 2. Q.E.D.

Corollary 1 *The digraph $\vec{G}(V^0)$, constructed by the algorithm VS, is vertex-saturated.*

Theorem 3 *VS-digraph can be constructed in time $O(n^5)$.*

One completion of the steps 1 – 6 requires $O(n_k^3)$ time units, where n_k is the size of the vertex set of $\vec{G}(V^k)$. Assuming that for each completion of these steps the maximum antichain increases at one vertex, we obtain the design time of a digraph $\vec{G}(V^k)$ vertex-saturated with respect to the initiating set, is equal to $O(n_k^4)$.

Hence, the design time of a correctly constructed vertex-saturated digraph $\vec{G}(V^0)$ is equal to:

$$\sum_{\forall n_k} O(n_k^4) = O(n^5).$$

Q.E.D.

Theorem 4 *Let a digraph $\vec{G}(V^0)$ be vertex-saturated. Then there exists an MCP \mathcal{P} of the graph $\vec{G}_t(V^0)$ such that its chains contain only essential arcs.*

Let a digraph $\vec{G}(V^0)$ is constructed by the algorithm VS. By Theorem 2, each bipartite digraph $G(V^k, V^{k+1})$ ($k = \overline{0, m}$) of this digraph satisfies the Hall's theorem and, hence, has a matching that saturates each vertex of the set V^{k+1} . Q.E.D.

Corollary 2 *Let $\vec{G}(V^0)$ be a VS-digraph. Then each chain of an MPP \mathcal{P} of $\vec{G}_t(V^0)$ is begun by some vertex v of V^0 .*

Due to this result we may use the adjacent matrix of $\vec{G}(V^0)$ as a working table for determination MCP of the TCG $\vec{G}_t(V^0)$ of a VS-digraph. Thus, we will suppose that chains of MPP of the TCG of a VS-digraph are found using the adjacent matrix of this digraph. **That is, we choose only essential arcs to construct each new MPP!**

Certainly, we use the adjacent matrix of the transitive closure graph to search for the maximum antichains $U(v)$ of such TCG.

The instance of constructing vertex-saturated digraph is shown in Appendix B.

4 An algorithm for finding MMIS of a graph

Let a saturated digraph $\vec{G}(V^0)$ is constructed, which has a MMIS \hat{U} such that $Card(\hat{U}) > Card(V^0)$. In this case, at least one of the chains of TCG of the VS-digraph $\vec{G}(V^0)$ contains a fictitious arc, whose endpoints belong to the MMIS.

Let, further, a some fictitious arc $v_i \gg v_j$ is found in the TCG $\vec{G}_t(V^0)$. We shall remove the vertices v_i, v_j from the digraph $\vec{G}_t(V^0)$ and all vertices,

which are adjacent with them. As a result, we shall obtain a digraph $\vec{G}_1(V_1^0) = (V_1, \vec{\Gamma}_1)$, where

$$V_1 = V \setminus (\{v_i, v_j\} \cup \Gamma(v_i) \cup \Gamma(v_j)),$$

$$V_1^0 = V^0 \setminus (\vec{\Gamma}^{-1}(v_i) \cup \vec{\Gamma}^{-1}(v_j)). \quad \vec{\Gamma}_1 = \vec{\Gamma} \cap V_1.$$

Here $\Gamma(v) = \vec{\Gamma}(v) \cup \vec{\Gamma}^{-1}(v)$.

For the digraph $\vec{G}_1(V_1^0)$, we shall use the procedure of constructing a VS-digraph by the algorithm VS. As a result, we shall obtain a digraph $\vec{G}(Z^0)$, which shall call *induced by removing the fictitious arc $v_i \gg v_j$* .

An algorithm for finding a MMIS of a digraph $\vec{G}(V^0)$ is constructed on the supposition that the following conjecture is true.

Conjecture 1 *Let a saturated digraph $\vec{G}(V^0)$ has an independent set $U \subset V$ such that $\text{Card}(U) > \text{Card}(V^0)$. Then it will be found a fictitious arc $v_i \gg v_j$ such that in the digraph $\vec{G}(Z^0)$, induced by removing this arc, the relation $\text{Card}(Z^0) \geq \text{Card}(V^0) - 1$ is satisfied.*

The worded conjecture allows to formulate a solution algorithm for finding a MMIS of a graph $G \in L_n$. Input of the algorithm is an undirected graph $G \in L_n$. Output of the algorithm is the MMIS.

An algorithm for finding a MMIS.

- Step 1.* Execute an initial orientation of the graph edges so to get an acyclic digraph $G(V^0)$.
- Step 2.* Execute the algorithm VS for the digraph $G(V^0)$.
- Step 3.* In TCG of the VS-digraph to find an unmarked fictitious arc $v_i \gg v_j$. Mark the found fictitious arc as considered. If all fictitious arcs are marked, go to the Step 7.
- Step 4.* Remove vertices v_i, v_j as well as all adjacent with them vertices from the saturated digraph $G(V^0)$. As a result, a digraph $\vec{G}_1(V_1^0)$ will be obtained.
- Step 5.* Execute the algorithm VS for the digraph $G_1(V_1^0)$. As a result, a digraph $\vec{G}(Z^0)$ will be obtained.
- Step 6.* If $\text{Card}(Z^0) \geq \text{Card}(V^0) - 1$, construct a set $W = Z^0 \cup \{v_i, v_j\}$ and execute the cutting operation $\sigma_W(\vec{G}(V^0))$ in the saturated digraph $G(V^0)$. Go to Step 2. Otherwise go back to Step 3.
- Step 7.* Put a MMIS $\hat{U} = V^0$.

Theorem 5 *If the conjecture 1 is true then the stated algorithm finds a MMIS of the graph $G \in L_n$.*

It is obviously.

Q.E.D.

Theorem 6 *The running time of the algorithm of finding a MMIS equals to $O(n^8)$.*

Indeed, single executing of the Steps 3 – 6 requires $O(n_1^5)$ of time units, where n_1 is the number of vertices in the digraph $\vec{G}(Z^0)$, induced by removing a fictitious arc. Since total number of fictitious arcs is $O(n^2)$, in worse case, for executing the Steps 3 – 6 is required $O(n^7)$ of time units. If suppose that after executing these steps, the found independent set V^0 will be increased to the unit, the total running time of steps 2 – 6 equals to $O(n^8)$. Q.E.D.

5 Conclusion

The pascal-programs were written for the proposed algorithm. Long testing the program for random graphs has shown that the algorithm runs stably and correctly.

Of course, the offered algorithm is not competitive in practice because of high degree of polynomial estimation of the running time. However, the algorithm has important theoretical significance since there is a good probability to prove that for NP-complete problems possible to construct a polynomial-time algorithm.

Author thanks Guenter Stertenbrink¹ for the help in testing the programs.

References

- [1] P. Crescenzi and V. Kann. A compendium of np optimization problems. Technical report, Royal Institute of Technology, Stocholm, 1998. This is the catalog of **NP** optimization problems. Also available at <ftp://ftp.nada.kth.se/Theory/Viggo-Kann/compendium.ps>.
- [2] L. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton Univ. Press, Princeton, N. J., 1962.
- [3] M. R. Garey and D. S. Johnson. *Computers and Intractability*. W.H.Freeman and Company, San Francisco, 1979.
- [4] J. E. Hopcroft and R. M. Karp. A $n^{5/2}$ algorithm for maximum matching in bipartite graphs. *J. SIAM Comp.*, 2:225–231, 1973.
- [5] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: Algorithms and Complexity*. Prentice-hall Inc., Engiewood Cliffs, N. J., 1982.
- [6] F. M. Reingold, J. Nivergelt, and N. Deo. *Combinatorial Algorithms (Theory and Practice)*. Prentice-hall Inc., Engiewood Cliffs, N. J., 1977.
- [7] M. N. S. Swamy and K. Thulasiraman. *Graphs, Networks and Algorithms*. John Wiley & Sons, N. Y., Chichester, 1981.
- [8] R. E. Tarjan. Complexity of combinatorial algorithms. *SIAM Rew.*, 20(3):457–491, 1978.

¹sktekrtken(at)akol.com (remove k's)

- [9] D. B. West. *Introduction to Graph Theory*. Prentice Hall, Inc., Upper Saddle River, NJ, 1996.

A Partially Ordered Sets

Relations between two objects are called *binary*. A binary relation R may be represented by a listing of object pairs, which are in the relation R :

If $(a, b) \in R$, then this fact we also denote by aRb . When $(a, b) \notin R$, then we will write $a\bar{R}b$. If $a \in A$ and $b \in A$ for all aRb , then R is called a *relation on the set A* . Further, we will consider only relations on the *finite* set A .

A relation R is *reflexive* if aRa for every $a \in A$. A relation R is **irreflexive** if $a\bar{R}a$ for every $a \in A$. A relation R is *symmetric* if whenever aRb , then bRa . A relation R is *antisymmetric* if whenever aRb and bRa , then $a = b$. A relation R is *transitive* if whenever aRb and bRc , then aRc .

(a)

(b)

In Fig. 2, the acyclic graph represents the partial order, induced by the binary relation

Here and throughout, we assume that the orientation of arcs of a digraph on a drawing is from below upwards.

12

Theorem 7 (Dilworth R.P.) *Let (A, \geq) be a finite partially ordered set. The minimum number of disjoint chains, which the set (A, \geq) can be partitioned on, equals to the capacity of the maximum antichain in (A, \geq) .*

There is an efficient algorithm for the partitioning a finite partially ordered set into the minimum number of chains and for finding the maximum antichain, elaborated by L. R. Ford and D. R. Fulkerson [2]. In essence, this algorithm finds the maximum matching in a bipartite graph $G^* = (X, Y, \Gamma^*)$. If a partially ordered set has n elements, then this graph contains $2n$ vertices and $Card(X) = Card(Y) = n$. An edge (x_i, y_j) joins two vertices $x_1 \in X$ and $y_j \in Y$ if and only if the corresponding elements $a_1, a_2 \in A$ of (A, \geq) are comparable.

In manual computations, we will use an adjacent matrix M of G^* as a working table. Units of M determine its *admissible cells*. Two cells of M are called *independent* if they are located in distinct rows and distinct columns of M . To find the maximum matching, we will have to find the maximum number of admissible independent cells of M .

The algorithm for partitioning a partially ordered set into the minimum number of chains consists of two stages:

- Construct an initial partition of the partially ordered set into chains;
- Improve the existing partition if it is possible.

To avoid ambiguity, we always look through rows and columns of M uniformly: from top to bottom in columns and from left to right in rows.

To obtain an initial partition, we may use the following procedure.

Step 0. Put $N = n$, where n is the number of rows M , $i = 1$.

Step 1. If $N = 0$, then complete the calculations as an initial partition is found.

Step 2. In i -th row of M find the first on the order admissible cell, whose appropriate column is not marked. If such cell is not found, put $i := i + 1$, $N := N - 1$ and go to Step 1. Otherwise, remember the found cell (i, j) , mark a column j , calculate $i := i + 1$ and go to Step 1.

Fig. 3 (a) shows the adjacent matrix for the strictly ordered set, represented in Fig. 2. The chosen cells of the initial partition are indicated by a circle.

To find a MCP and the maximum antichain of a set (A, \geq) , we will make use of the Ford-Fulkerson's algorithm [2]. The algorithm begins to work after termination of the previous procedure, that is, when there exists an initial partition of the ordered set into chains.

Step 1. Mark rows of M that do not contain the chosen cells, by the symbol $(*)$.

Step 2. Look through the newly marked rows of M and find all unchosen cells in each row. Mark all unmarked columns of M that correspond with such cells by an index of the row.

- Step 3.* Look through the newly marked columns. If an examined column contains a chosen cell (that is, the cell is enclosed within a circle), then mark the row containing the chosen cell by an index of the examined column. If the column does not contain a chosen cell, go to Step 4. If it is impossible to mark new rows, then go to Step 5.
- Step 4.* The essence of the given step is the procedure of constructing a new collection of independent cells, each having one more cell than the former collection. At each stage of this procedure, except for the final step, we pick a *new* admissible cell of M and delete the “old” one. Increasing the total number of chosen cells happens as follows. In the found j -th column, choose a new cell in a row $m(j)$, where $m(j)$ is a mark of the current column. Let we already have chosen the cell (i, j) , which marks $m(i)$ and $m(j)$ correspond to, where $m(i)$ is a mark the i -th row. If $m(j) = (*)$, the procedure of constructing a new collection of independent cells is completed. Delete all marks of rows and columns, and go to Step 2. Otherwise, delete the cell $(i, m(i))$ and choose a cell $(m(m(i)), m(i))$. Put $i = m(m(i))$, $j = m(i)$ and repeat the process described above.
- Step 5.* Find the maximum antichain $\mathcal{U} = U_r \setminus U_c$, where U_r is a set of marked rows, and U_c is a set of marked columns. Terminate the calculations. The found admissible cells determine arcs forming chains of the MCP.

Fig. 3 (b) shows the picked cells of the optimal partition for the partially ordered set, represented in Fig. 2. In this case, the MCP consists of the chains $A_1 = \{a_1, a_3\}$, $A_2 = \{a_4, a_2\}$, and $A_3 = \{a_5\}$. We also have a set $U_r = \{a_2, a_3, a_4, a_5\}$ of marked rows, and a set $U_c = \{a_2\}$ of marked columns. Consequently, the maximum antichain \mathcal{U} is equal to

$$\mathcal{U} = \{a_2, a_3, a_4, a_5\} \setminus \{a_2\} = \{a_3, a_4, a_5\}.$$

Ford-Fulkerson’s methodology, described above, for finding antichains may be easily adapted to any algorithm of finding the maximum matching in a bipartite graph, for example, Hopcroft-Karp’s algorithm [4], [7], or flow algorithm [5]. Therefore, we assume that the running-time of a MCP construction is equal to $O(n^{5/2})$.

B An instance of constructing a vertex-saturated digraph

Consider an instance of constructing a vertex-saturated digraph (VS-digraph).

Fig. 4 shows a digraph $\vec{G}(X^0)$, obtained from an initial undirected graph G as it was described in part 2. Recall that the orientation of arcs of the digraph in figures is from below upwards.

Construct a VS-digraph.

The adjacent matrix of the TCG $\vec{G}_t(V^0)$ is shown in Fig. 5. The fictitious arcs of this graph are represented by the letter **f**. Arcs, belonging to the MCP of $\vec{G}_t(V^0)$, are put into circles. These arcs are shown in Fig. 4 by thick lines.

First of all, notice that the initiating set $V^0 = \{v_1, v_2, v_3, v_4\}$ is a MIS of $\vec{G}(V^0)$.

Find the general antichain \mathcal{U} (see Fig. 5 (a)). The set of marked rows is $U_r = \{5, 7, 8, 10\}$, and the set of marked columns is empty, that is, $U_c = \emptyset$. Therefore:

$$\mathcal{U} = U_r \setminus U_c = \{5, 7, 8, 10\} \setminus \emptyset = \{5, 7, 8, 10\}.$$

This antichain is a MIS of the digraph, and $Card(\mathcal{U}) = Card(V^0)$.

Find the maximum antichains $\mathcal{U}(v)$ for graph vertices.

Clearly, $\mathcal{U}(v_5) = \mathcal{U}(v_7) = \mathcal{U}(v_8) = \mathcal{U}(v_{10}) = \mathcal{U}$.

To find $\mathcal{U}(v_1)$, mark the first row of the adjacent matrix of $\vec{G}_t(X^0)$.

Marking the first row in Fig. 5 (b), we have $U_r = \{1, 3, 4, 5, 6, 7, 8, 9, 10\}$, and $U_c = \{5, 7, 8, 9, 10\}$. Consequently,

$$\mathcal{U}(v_1) = U_r \setminus U_c = \{1, 3, 4, 6\}.$$

This antichain is a MIS of the digraph, and $Card(\mathcal{U}(v_1)) = Card(X^0)$.

Similarly, marking the second row in Fig. 5 (c), we have

$$\mathcal{U}(v_2) = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \setminus \{5, 6, 7, 8, 9, 10\} = \{1, 2, 3, 4\}.$$

This maximum antichain is also a MIS of the digraph, and $Card(\mathcal{U}(v_2)) = Card(V^0)$.

Similarly, we obtain the maximum antichains

$$\mathcal{U}(v_3) = \{3, 4, 5, 8\},$$

$$\mathcal{U}(v_4) = \{4, 5, 8, 9\},$$

and

$$\mathcal{U}(v_6) = \{3, 4, 5, 6\}$$

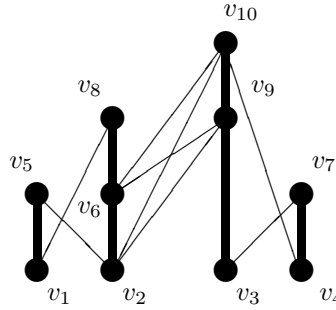


Figure 4: A digraph

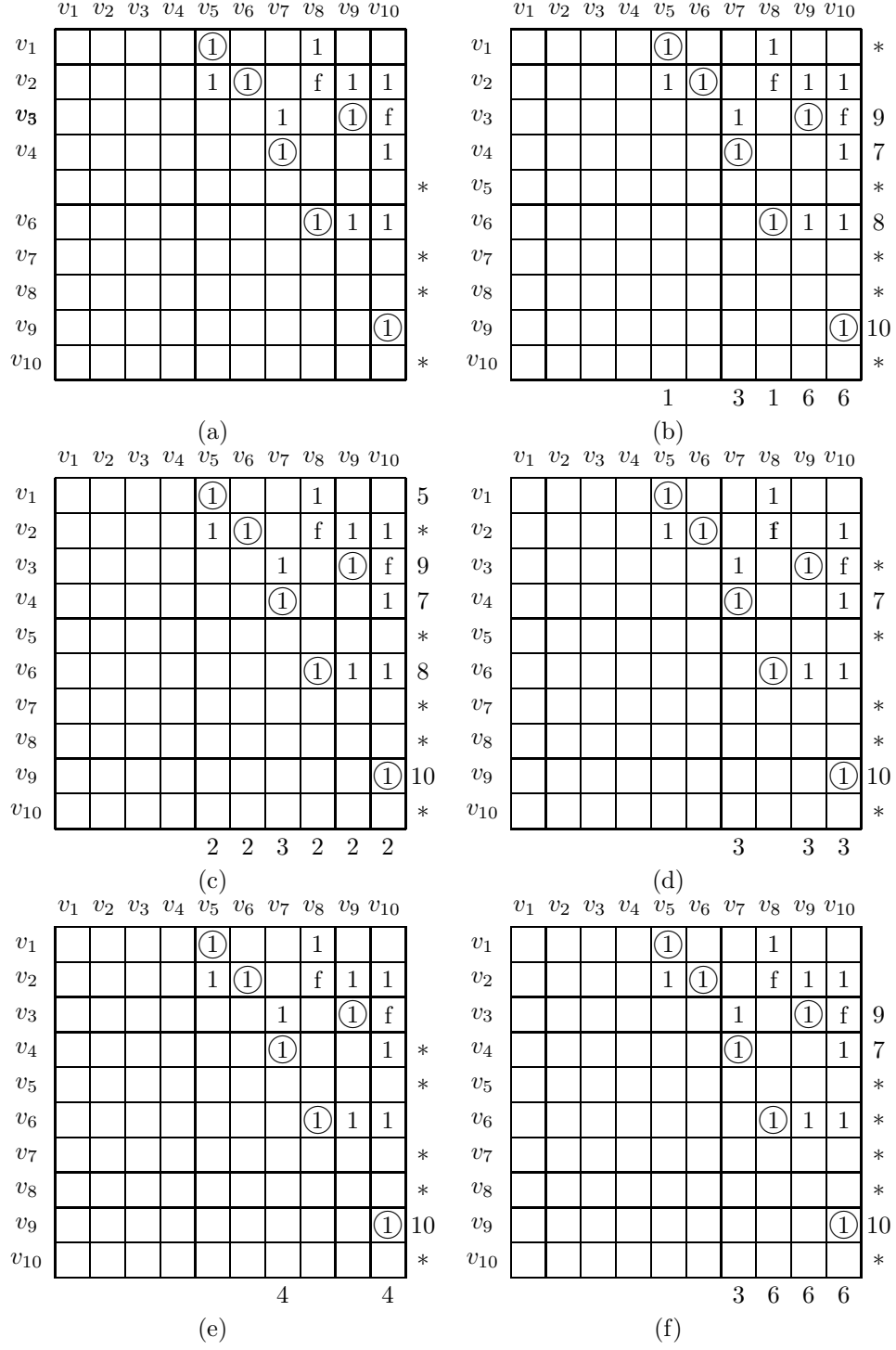


Figure 5: Finding antichains

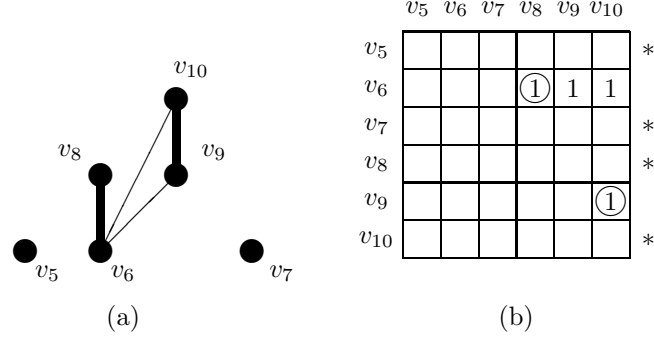


Figure 6: The directed subgraph induced by the layer V^1

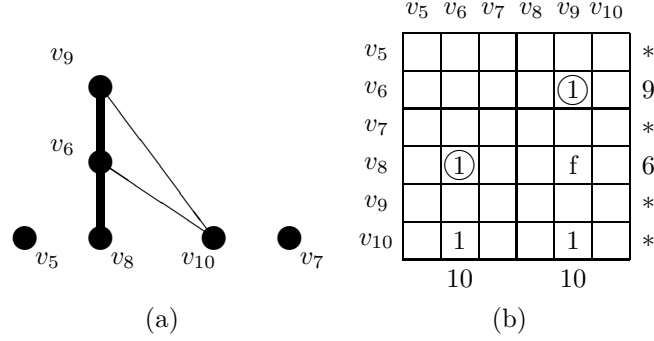


Figure 7: The new directed subgraph

from Fig. 5 (d), (e), and (f) correspondingly. Each of these maximum antichains is a MIS of the digraph $\vec{G}(V^0)$, and they have the number of elements equal to V^0 .

Thus, the digraph $\vec{G}(V^0)$ is saturated with respect to the initiating set V^0 .

Now we examine directed subgraphs induced by layers of $\vec{G}(V^0)$.

Consider directed subgraph $\vec{G}(V^1)$ induced by layer $V^1 = \{v_5, v_6, v_7\}$. This subgraph is represented in Fig. 6 (a). Notice that the adjacent matrix of the TCG $\vec{G}_t(V^1)$ can be obtained from the adjacent matrix of $\vec{G}_t(V^0)$ directly. Obviously, the MCP of $\vec{G}_t(V^1)$ is a part of the MCP of $\vec{G}_t(V^0)$. The adjacent matrix of $\vec{G}_t(V^1)$ is shown in Fig. 6 (b). The general antichain \mathcal{U}_1 of $\vec{G}_t(V^1)$ equals

$$\mathcal{U}_1 = \{v_5, v_7, v_8, v_{10}\}.$$

This antichain is a MIS of $\vec{G}(V^1)$; however, $\text{Card}(\mathcal{U}_1) > \text{Card}(V^1)$. Consequently, the directed subgraph $\vec{G}(V^1)$ is not vertex-saturated with respect to its initiating set.

Therefore, we assume $W = \{v_5, v_7, v_8, v_{10}\}$ and reorientate all arcs of $\vec{G}(X^1)$ incoming to the vertices of W . We obtain a new directed subgraph represented in

Fig. 7 (a). Clearly, this subgraph has a new initiating set $V^1 = \{v_5, v_7, v_8, v_{10}\}$.

Examining as above, we find that the new directed subgraph $\vec{G}(V^1)$ is vertex-saturated with respect to its initiating set V^1 .

Thus, we may construct a new digraph $\vec{G}(V^0)$. The adjacent matrix of this digraph can be obtained from the adjacent matrix of the initial digraph if the corresponding part of it is replaced by the adjacent matrix of $\vec{G}(V^1)$.

Similarly, we determine that a directed subgraph $\vec{G}(V^2)$, where $V^2 = \{6\}$, is vertex-saturated with respect to its initiating set V^2 .

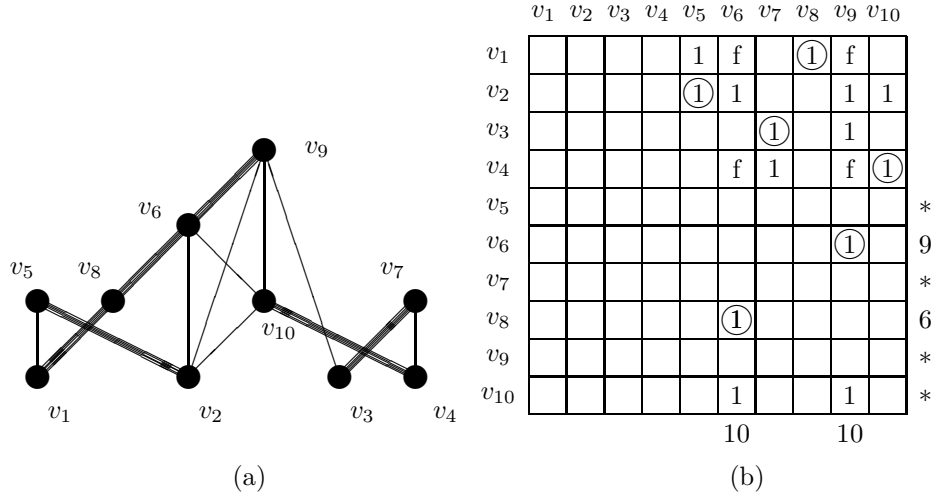


Figure 8: The VS-digraph

At last, we may make sure that the new digraph $\vec{G}(V^0)$ is a VS-digraph since each of its directed subgraphs is vertex-saturated with respect to its initiating set. This digraph is represented in Fig. 8 (a). The adjacent matrix of the transitive closure graph $\vec{G}_t(V^0)$ and its MCP together are represented in Fig. 8 (b).